10/756,123 PTO-892

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(54) Title: PIPELINE FLOW CONTROL OPTIMIZATION SOFTWARE, AND METHODS

DELIVERY △ SUPPLY ● Station ▷ Station Control Logic
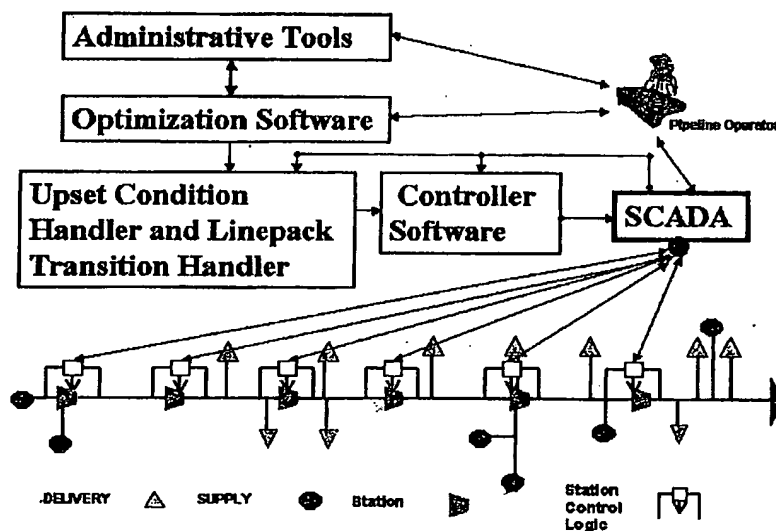
(57) Abstract: Pipeline flow optimization systems, software, and methods that emulate transient optimization are disclosed. A pipeline flow optimization system includes an upset condition handler that is adapted to provide a multivariable controller with pipeline flow adjustments to handle one or more upset conditions. The optimization system may be used with fluid pipeline systems, including gas pipeline systems and liquid pipeline systems. Systems, software, and methods according to embodiments of the invention may further include linepack transition handling capabilities and administrative tools.

WO 2006/014372 A2

WO 2006/014372 A2

PIPELINE FLOW CONTROL OPTIMIZATION SOFTWARE AND METHODS

BACKGROUND OF THE INVENTION

1. Field of the Invention

[0001] The invention relates to software and methods for fluid pipeline optimization and control, and more specifically, to optimization and control software and methods that can optimize for transient conditions within the fluid pipeline.

2. Description of Related Art

[0002] Fluid pipeline systems, such as gas transmission pipelines, operate at certain pressures and with certain flow rates in order to deliver fluid, for example, natural gas, to its destination. In the case of a gas pipeline, compressors along the pipeline maintain the pressures necessary to move the gas. In the case of a liquid pipeline, pumps move the liquid along the pipeline.

[0003] FIGURE 1 is a diagram showing a typical gas pipeline system. Generally, referring to FIGURE 1, a typical gas pipeline system may include multiple relay compressor stations with multiple points of supply and deliveries. For example, FIGURE 1 shows a pipeline system having 6 compressor stations. Each individual station is controlled by control logic that, among other things, operates compressors and prevents a station from exceeding its maximum allowable operating pressures. The pipeline operator, through a supervisory control and data acquisition (SCADA) system, gives each individual compressor station's control logic its own target setpoints for suction pressure, discharge pressure and flow. The station's control logic seeks to maintain these target setpoints by starting and stopping compressors, as necessary. Additionally, as noted above, the control logic protects the station from exceeding its maximum allowable operating pressures and maintains it

5   within safe operating parameters. However, the station control logic controls an individual

station. Hence, it functions independent of the other stations in the system.

[0004]   FIGURE 2 is a diagram showing a pipeline flow control system that may be

utilized to control the pipeline system of FIGURE 1. Generally, referring to FIGURE 2, a

pipeline flow control system may include steady-state Optimization Software that calculates

10   optimal target setpoints for the pipeline operator to manually send to the individual stations'

control logic. Further, as FIGURE 2 shows, the system may also include a multivariable

control system ("Controller Software") operating in conjunction with the optimization

software. The multivariable controller software may include a set of controllers that attempts

to drive a set of station discharge pressures to equal a set of discharge pressure targets

15   calculated by the optimization software, and there may be one controller for each compressor

station that is operating under multivariable control. Each controller attempts to manipulate

the suction pressure of the compressor station it controls in order to drive the discharge

pressure of the next upstream station toward its optimal discharge pressure target.

[0005] However, fluid pipelines are dynamic systems, and the control systems

20   described with reference to FIGURE 1 and FIGURE 2 may not anticipate or take into account

certain transient (i.e. non-steady state) conditions. Additionally, because each controller in

the system of FIGURE 2 typically controls its station in light of operating conditions at that

station and the next upstream station only, the controller software is controlling individual

stations on the pipeline, which are all interconnected, based on data gleaned from only a

25   portion of the stations in the system. Hence, the controller software may not effectively

control all stations on the pipeline system in light of transient conditions affecting the entire

pipeline system or other portions of the pipeline system beyond the control of an individual

5   controller. Accordingly, the controller software may, among other things, increase the cost of

transporting fluid through the pipeline.

[0006] Further, the system shown in FIGURE 2 utilizes a single set of tune

parameters to handle all transient conditions but that are aimed at only controlling the

setpoints of the immediate upstream compressor station. However, certain transient

10   conditions may occur in the pipeline system that can not be efficiently or effectively managed

by only controlling the setpoints of the immediate upstream compressor station. In such

transient conditions, the controller software's single set of tune parameters does not provide

the capability to adjust station operating parameters to effectively or timely handle the

transient conditions. Hence, in this case, the stations' control logic over-rides the

15   multivariable controller software and controls corresponding individual compressor station

targets to near maximum operating pressures. Generally, this leads to an imbalance in the

system, which usually results in not being able to take the full contracted supplies into the

transmission system.

[0007] Additionally, the control system of FIGURE 2 is not able to identify or

20   transition to a new optimal solution when conditions change (e.g. supplies or deliveries of the

transmission system) that require a change to the optimal compressor configuration. A

particular problem encountered was when to start additional compressors when required for

an optimal fuel-efficient system. Starting the additional compressors too early wastes fuel.

On the other hand, starting the additional compressors too late may cause a transient

25   condition that is beyond the capability of the controller software to control, resulting in

station control logic overriding the controller software.

SUMMARY OF THE INVENTION

[0008] The present invention provides a pipeline flow optimization system that emulates transient optimization.

[0009] Additional features of the invention will be set forth in the description which follows, and in part will be apparent from the description, or may be learned by practice of the invention.

[0010] The present invention discloses a pipeline flow optimization system, comprising an upset condition handler adapted to provide a multivariable controller with pipeline flow adjustments to handle one or more upset conditions.

[0011] The present invention also discloses a pipeline flow optimization system including an upset condition handler, a multivariable controller for controlling components of a pipeline system, optimization software for calculating steady-state parameters of the pipeline system and providing the steady-state parameters to the upset condition handler, and a supervisory control and data acquisition system for acquiring pipeline system data. The upset condition handler receives the pipeline system data from the supervisory control and data acquisition system and provides the multivariable controller with pipeline flow adjustments to handle one or more upset conditions.

[0012] The present invention also disclose a pipeline flow control system including an optimizer for calculating steady-state parameters of a pipeline system, a controller for controlling pipeline system components that cause pipeline flow, a data acquirer for acquiring pipeline system component data and pipeline flow data, and a control logic unit. The control logic unit adjusts the steady-state parameters of the pipeline system in response to an upset condition by analyzing the pipeline system component data and the pipeline flow data and applying an adjustment factor to the controller.

[0013] The present invention also discloses a method of handling transient conditions in a pipeline flow optimization system. In the method, one or more adjustment factors are calculated. Each of the adjustment factors is indicative of a degree to which a correction is needed for an upset condition. The adjustment factors are stored, one or more of the adjustment factors are chosen and applied to the pipeline flow optimization system.

[0014] It is to be understood that both the foregoing general description and the following detailed description are exemplary and explanatory and are intended to provide further explanation of the invention as claimed.

BRIEF DESCRIPTION OF THE DRAWINGS

[0015] The accompanying drawings, which are included to provide a further understanding of the invention and are incorporated in and constitute a part of this specification, illustrate embodiments of the invention and together with the description serve to explain the principles of the invention.

[0016] FIGURE 1 is a diagram showing a typical gas pipeline system.

[0017] FIGURE 2 is a diagram showing a pipeline flow control system that may be utilized to control the pipeline system of FIGURE 1.

[0018] FIGURE 3 is a diagram showing an automated pipeline optimization system according to an embodiment of the invention.

[0019] FIGURE 4, FIGURE 5, FIGURE 15, FIGURE 16, and FIGURE 17 are schematic flow diagrams illustrating a pipeline flow management system in accordance with embodiments of the invention.

[0020] FIGURE 6 and FIGURE 7 are schematic flow diagrams illustrating the general process of computing and applying adjustment factors for a number of upset conditions.

[0021] FIGURE 8 and FIGURE 12 are schematic flow diagrams illustrating the computation of adjustment factors for a number of upset conditions.

[0022] FIGURE 9, FIGURE 10, FIGURE 11, and FIGURE 13 are schematic flow diagrams illustrating expressions used to compute adjustment factors for various upset conditions.

[0023] FIGURE 14 is a schematic flow diagram illustrating the application of the computed adjustment factors to the pipeline flow control optimization system.

[0024] FIGURE 18 and FIGURE 19 show steps for saving a revised current nomination and a future nomination, respectively, and FIGURE 20, FIGURE 21, FIGURE 22, FIGURE 23, FIGURE 24, and FIGURE 25 are flow charts illustrating a transition process according to an embodiment of the invention.

[0025] FIGURE 26 shows general calibration processes that may be completed each day, and FIGURE 27, FIGURE 28, FIGURE 29, FIGURE 30, FIGURE 31, and FIGURE 32 are schematic flow diagrams illustrating tasks involved in calibrating the system according to an embodiment of the invention.

[0026] FIGURE 33, FIGURE 34, FIGURE 35, and FIGURE 36 are flow charts illustrating steps of a method for accurately measuring the fuel savings of the system according to an embodiment of the invention.

[0027] FIGURE 37 shows fuel attainment variances that may be taken into account when measuring the fuel savings of the system according to an embodiment of the invention.

DETAILED DESCRIPTION

[0028] As was described above, aspects of the invention relate to systems and methods for fluid pipeline flow optimization and, in particular, systems and methods for fluid pipeline flow optimization that are capable of handing transient or upset conditions. In their broadest embodiments, software and methods according to the invention ensure that the demand for a fluid flowing in a pipeline is met. That demand is typically subject to certain requirements, for example, that particular volumes of the fluid be delivered at pressures between specified minimum and maximum delivery pressures. Additionally, systems and methods according to the invention ensure that fluid pipeline transmission facilities are operating within the engineering design parameters for which they were designed. In general, those two requirements are met by regular monitoring and by controlling fluid pipeline parameters to keep them within acceptable tolerances (sometimes referred to as "deadbands"). Systems and methods according to embodiments of the invention may also provide human controllers with feedback as to the pipeline's present fuel efficiency, and may provide engineers with calibration tools necessary to tune software, systems, and methods according to the invention.

[0029] Software according to embodiments of the invention provides one or more of the following three modules or capabilities: upset condition algorithms and upset condition handling, linepack transition handling, and administrative tools. Upset condition algorithms recognize when conditions vary from the static optimization and change compressor setpoints along the gas pipeline to account for the upset while continuing to optimize fuel efficiency within the given compressor unit configuration. Linepack transition handling algorithms

7

5 calculate the time required to attain optimal pressure targets when a future optimal run

dictates a change in compressor unit configuration. Administrative tools comprise

measurement methods used to determine fuel efficiency performance, and allow accurate cost

savings reports to be generated. Administrative tools may also include calibration tools used

to tune the models created by the software.

10   [0030] Although most aspects and embodiments of the invention are equally

applicable to any type of fluid pipeline system, including gas pipelines and liquid pipelines,

certain embodiments of the invention will be described with respect to gas transmission

pipeline networks.

      [0031] FIGURE 3 shows an automated pipeline optimization (APO) system

15 according to an embodiment of the invention. Referring to FIGURE 3, systems and methods

according to gas pipeline-based embodiments of the invention typically combine a

multivariable control system ("Controller software") with optimization simulation

programming ("Optimization Software") through a supervisory control and data acquisition

(SCADA) system to reduce fuel consumption at compressor stations along the gas pipeline

20 network. The APO system utilizes an upset condition handler and a linepack transition

handler ("Black Box") to optimize pipeline operations by accounting for and handling all

upset conditions and linepack transitions. The upset condition handler handles transient

conditions, including those that previously would have caused the stations' control logic to

override the controller software. The APO system detects transient conditions occurring

25 throughout the pipeline system, evaluates and selects a transient condition to address, and

then chooses an optimal solution, from among multiple potential solutions, to address the

selected condition.

[0032]   Accordingly, utilizing the Administrative Tools, the Upset Condition
Handler, and the Linepack Transition Handler in the pipeline flow optimization system of
FIGURE 3 provides a system that may maintain optimal steady-state configurations and
properly adjust optimal steady-state pressure targets through upset conditions; that may
identify through existing field compressor logic, or supply or delivery change of the
predictive models, when a new compressor configuration may be required; that may quickly
transition to the new optimal solution; that may verify results through a verification process
of an Automated Calibration Tool; and that may quantify the results of the pipeline
optimization process through a Fuel Attainment process.

[0033]   In embodiments of the invention, the methods according to the invention may
be incorporated into the multivariable control system, or they may be implemented
separately, as shown in FIGURE 3, in which case they would be used to generate inputs into
the multivariable control system. Although any multivariable control system may be used,
GE-CCI's MVC-4.0® multivariable control system is one suitable system. Further, although
any optimization software may be used, Advantica's Stoner Gas Solver for EMM (Energy
Minimization Module) is one suitable package. Whether alone or as part of a multivariable
control system, systems, methods and software according to embodiments of the invention
may be programmed and implemented in any one of a number of programming languages,
including C, C++, C#, Java, Visual Basic, and any other programming language able to
implement the tasks of the methods.

[0034]   One embodiment of the invention was implemented for testing using a section
of gas transmission line including approximately 220 miles of gas transmission piping with
six compressor stations spaced 40-45 miles apart along the gas transmission line, and certain
portions of the description that follows will refer to specific locations along that testbed gas

5   transmission line. However, it should be understood that systems, methods, and software

according to embodiments of the invention may be implemented in fluid pipeline systems of

any size or complexity.

[0035] FIGURE 4, FIGURE 15, FIGURE 16 and FIGURE 17 are schematic flow

diagrams showing the overall control logic used to select the mode of the system. Each time

10   the system is turned on, the total available horsepower is summed, and other power-on and

initialization tasks may be performed. (Several exemplary tasks are shown in FIGURE 4).

Once the initialization tasks are performed as per FIGURE 4 and FIGURE 5, the control logic

(i.e. the upset condition handler and the controller software) monitors the operating mode of

the system. During times when the system is operating normally, the system is kept in

15   "normal" mode. In "normal" mode, the system can handle transient conditions and changes

within certain tolerances.

[0036] If a severe problem should occur in the pipeline, such as the unexpected

shutdown of a compressor that is needed to maintain system integrity, the control logic

changes the mode of the system for that particular compressor to "event." In "event mode,"

20   the operator is notified and the control logic that would normally allow compressors to

operate at or near 100% of their rated power without starting new compressors is deactivated

so as to facilitate starting replacement compressors. In addition, appropriate pressure set

points upstream of the station in "event" mode are reduced to "event" mode settings to

minimize the probability that the upstream station will shut down because of over-pressure

25   alarm conditions. The appropriate target pressures downstream from the station in "event"

mode are reduced by the algorithms described above and below. The operator may also take

additional actions to resolve the problem. When "event mode" is terminated, the system

resets the settings to conform to "normal mode" settings. As compared with standard

5   multivariable control systems, "event" mode in systems according to the present invention is

typically invoked only selectively and only to the extent needed to maintain system integrity.

[0037] In both "normal" and "event" modes, the system uses a factor- or parameter-

based method for dealing with upset or transient conditions. This method is generally

illustrated in the schematic flow diagram of FIGURE 6. The actual upset or transient

10 conditions that the system detects and handles may vary from pipeline to pipeline, and are

generally defined depending on the type of pipeline and the needs of the users. However, the

parameter-based method will be described with respect to certain standard upset conditions

for a gas pipeline.

[0038] As shown in FIGURE 6, in general, the system initializes an array or other

15 data structure. In other words, the system clears out old data and inputs current data. Prior to

initializing the array or data structure, the system determines whether to adjust a flow target,

as shown in FIGURE 6 and FIGURE 7. Each data location within the array or other data

structure contains a value for an adjustment factor indicative of the degree to which a

correction is needed for a particular upset condition. The system engages in an iterative

20 process of calculating the values of each of the adjustment factors and storing them in the

array or other data structure. Once values for each adjustment factor have been calculated,

the system selects and applies the most necessary of the adjustment factors. Typically, the

most necessary of the adjustment factors would be the adjustment factors with the largest

values in the array or other data structure, although this need not be the case.

25 [0039] For example, in FIGURE 6, the array contains space for the values of ten

adjustment factors, indexed 1-10, although as many adjustment factors as desired may be

monitored by systems, methods, and software according to the invention. It should also be

understood that the upset conditions and corresponding adjustment factors may include both

5   adjustment factors for which the system can take automatic action to correct and adjustment

factors for which the only action is to notify the human operator of the condition. This will

be described below in more detail.

[0040] FIGURE 8 and FIGURE 12 are schematic flow diagrams illustrating the

iterative calculation of each of the adjustment factors for the respective upset condition. As

10   shown in FIGURE 8, the first upset condition (j=1) is the condition in which the flow at the

intake ("Station #1" in the figure) is less than the target flow rate. The second upset

condition (j=2) is the condition in which an upstream compressor station is in a high-flow

transient condition. The third upset condition (j=3) is the condition in which one or more of

the compressor station discharge pressures is close to or at the maximum operating pressure

15   ("MOP" in the figure). The fourth upset condition (j=4) is the upset condition in which the

throttle or speed driver of one or more compressors is near 100%. The fifth upset condition

(j=5) is the upset condition in which the throttle or power of one or more compressors is near

0%. The sixth upset condition (j=6) is the upset condition in which the cooler temperature is

close to maximum. The seventh upset condition (j=7) is the upset condition in which an

20   engine needs to start in a station that is already running. Referring to FIGURE 12, the eighth

upset condition (j=8) is the upset condition in which a discharge pressure at a particular

location ("Station #5" in the figure) is close to a delivery pressure required by a customer (

"Customer #1" in the figure). The ninth upset condition (j=9) is the upset condition in which

the supply pressure at a particular point in the pipeline ("Supplier #3" in the figure) is close to

25   the pressure in the pipeline. The tenth upset condition (j=10) is the upset condition in which

the delivery pressure at a particular point in the pipeline ("Customer #2 delivery pressure" in

the figure) is close to the pressure in the pipeline. The eighth, ninth, and tenth upset

conditions are conditions that the multivariable control system may not be able to directly

address, so action on those three upset conditions is limited to notifying the human operator.

12

5   However, as was noted above, the array of adjustment factors for upset conditions may

include any number of factors and may include both factors for upset conditions that can be

handled automatically and factors that require operator notification in any combination. In

some embodiments, only factors that can be handled automatically may be included in the

array or other data structure. As shown in FIGURE 6, FIGURE 8 and FIGURE 12, once

10   adjustment factors are calculated for each of the ten upset conditions defined in this

embodiment of the invention, the appropriate adjustment factors are chosen and applied.

[0041] FIGURE 9 is a schematic flow diagram illustrating the calculation of the first

through third adjustment factors (j=1, j=2, and j=3). As was described above, the first

adjustment factor relates to the upset condition in which the flow at the intake point in the

15   pipeline is less than the target flow. In this upset condition, it is assumed that the gas supplier

will eventually increase the flow into the pipeline so as to meet their contractual obligations.

Therefore, in the case of low flow at the intake, systems according to the invention lower the

discharge pressure target at the intake. This creates capacity in the system downstream to

accommodate the gas that will be pumped into the system later. It also reduces the stress on

20   the compressors as the gas is added at increased flow rates later. The expression used to

calculate the first adjustment factor is shown in FIGURE 9. Descriptions of the variables

used in FIGURE 9 are given below in TABLES 1 and 2.

[0042] As was described above, the second adjustment factor relates to the upset

condition in which the discharge pressure is greater than the target discharge pressure. When

25   the discharge pressure is greater than the target discharge pressure, systems according to

embodiments of the invention lower the target discharge pressure at downstream compressor

stations. This causes downstream compressors to speed up faster and reduces excess

5   linepack. Essentially, it improves the response time of the system. The expression used to

calculate the adjustment factor is shown in FIGURE 9.

[0043] The third adjustment factor, the calculation of which is also shown in FIGURE

9, relates to the upset condition in which a discharge pressure at one of the compressor

stations is close to the maximum operating pressure in the pipeline. As those of skill in the

10   art will appreciate, having the discharge pressure at a compressor station at a pressure close to

the maximum operating pressure is generally an undesirable condition. Therefore, in this

condition, the system lowers the target discharge pressure at downstream compressor stations

so that the downstream compressors will "pull" the pressure away. The expression used to

calculate the adjustment factor is shown in FIGURE 9.

15   [0044] FIGURE 10 is a schematic flow diagram illustrating the calculation of the

fourth, fifth, and sixth adjustment factors (j=4, j=5, and j=6). As was described above, the

fourth adjustment factor relates to the upset condition in which the throttle or speed driver of

one or more of the compressors is close to 100%. Therefore, in this condition, the system

lowers the target discharge pressure downstream, which reduces the speed driver or throttle

20   of the downstream compressors and allows the supply pressure to control the pressure in the

pipeline. The expression used to calculate the adjustment factor is shown in FIGURE 10.

Descriptions of the variables used in FIGURE 10 are given below in TABLES 1 and 2.

[0045] The fifth adjustment factor relates to the upset condition in which the throttle

or speed driver of one or more of the compressors is close to 0%. When the speed driver is

25   close to 0%, the discharge pressure is constrained and will not be able to decrease. In this

condition, the system lowers the discharge pressure target, which creates more available

capacity downstream in the pipeline. Then if a compressor station down stream should crash,

there will be sufficient time to recover before the station near 0% speed driver also crashes

14

5   due to over pressure. The expression used to calculate the adjustment factor is shown in

FIGURE 10.

[0046]   The sixth adjustment factor relates to the upset condition in which the cooler

temperature is close to maximum. When the cooler temperature is close to maximum, the

suction pressure is constrained and will not be able to decrease. In this condition, the system

10   lowers the discharge pressure target, which reduces the compression ratio and lowers the

discharge temperature unconstraining the suction pressure. The expression used to calculate

the adjustment factor is shown in FIGURE 10.

[0047]   FIGURE 11 is a schematic flow diagram illustrating the calculation of the

seventh adjustment factor (j=7). As was described above, the seventh adjustment factor

15   relates to the upset condition in which an engine needs to start in a station that is already

running. When an engine needs to start in a station that is already running, the station

controller will not allow the engine to start if the discharge pressure is greater than the limit

for starting engines. In this condition, if the discharge pressure is greater than the limit for

starting an engine, the system reduces the discharge pressure in time to start the engine. The

20   expression used to calculate the adjustment factor is shown in FIGURE 11.

[0048] FIGURE 13 is a schematic flow diagram illustrating the calculation of the

eighth, ninth, and tenth (j=8, j=9, and j=10). The eighth adjustment factor relates to an upset

condition for which automatic handling is not available, although in some embodiments, the

system might handle the upset condition automatically, at least to some extent. In the upset

25   condition to which the eighth adjustment factor relates, the pressure at a particular point in

the pipeline ("Station #5", as identified in FIGURE 13) is close to the limiting pressure

required by a customer ("Customer #1", as identified in the figure). (More particularly, the

algorithm queries whether a particular transmission line is flowing to the supplier and, if so,

15

5 to what degree a valve is open.) If the condition is true, a warning such as, "Customer #1

pressure close to Station #5's discharge header #2 pressure. Either change Station #5 loop

configuration or raise downstream suction pressure," may be provided. The expressions

used to calculate the eighth adjustment factor are shown in FIGURE 13. The ninth and tenth

adjustment factors also relate to pressure and flow upset conditions within the pipeline that

10 might make it difficult for the pipeline to supply a particular flow rate of gas at a particular

pressure so as to meet contractual obligations. In each case, the operator is provided with a

warning similar to that described above, and appropriate action is taken. The expressions

used to calculate the ninth and tenth adjustment factors are shown in FIGURE 13.

[0049] As those of skill in the art will note, the first seven adjustment factors deal

15 with transient conditions in the pipeline that can be controlled by the pipeline control system,

whereas the eighth, ninth and tenth adjustment factors deal with upset conditions that are

outside the scope of control of the pipeline control system. Although this distinction applies

in the present example and may not apply in all embodiments of the invention, the capacity of

software, systems, and methods according to the invention to indicate possible problems in

20 the pipeline that are beyond the scope of the multivariable control system logic to handle may

be a beneficial feature. Additionally, as was noted above, the multivariable control system

may also be programmed to handle the kinds of upset conditions represented by the eighth,

ninth and tenth adjustment factors. Furthermore, the additional adjustment factors may be

factors required for or desirable in specific pipelines or in specific configurations of software,

25 systems and methods according to the invention.

Table 1

APO: Automated Pipeline Optimization
Control Software Variables List
24JUN2005

Variables List 1

Variables:	Description:
Q	Flow Rate value into Station #1 (MMCFD)
FT	Flow Target now into Station #1 (MMCFD)
FFT	Future Flow Target into Station #1 (MMCFD)
c	# of Compressor Station: Station #1=0, Station #2=1, Station #3=2, Station #4=3, Station #5=4, Station #6=5
s	# of Downstream Compressor Station whose Ps is manipulated by Control Loop
t	# of Upstream Compressor Station whose Target Discharge Pressure is controlled by Control Loop
Event_Offset	Discharge Pressure Target Offset for stations upstream of a station in Event Mode.
TN	Current Time (Microsoft Excel Datetime format current time value adjusted to MMC time)
TS	Time Start: Time future flow target becomes effective (Datetime format)
TB	Time Flow Target is computed to Begin increasing from prior value to future value (Datetime format)
TE	Time Flow Target is computed to reach future value (Datetime format)
LT	Lead Time (Dead Band)

i	Value (Control Loop)	t	Controlled Discharge Pressure Target	s	Manipulated Suction Pressure
0		0	Station #1	1	Station #2
1		0	Station #1	2	Station #3
2		1	Station #2	2	Station #3
3		1	Station #2	3	Station #4
4		2	Station #3	3	Station #4
5		2	Station #3	4	Station #5
6		3	Station #4	4	Station #5
7		3	Station #4	5	Station #6
8		4	Station #5	5	Station #6

j	Value	Adjustment Factor: Description
0		Adjust Flow Target for New Nomination
1		Compensate for Station #1 flow less than target
2		Compensate for upstream station in high flow transient condition
3		Move station discharge pressures away from MOP
4		Reduce discharge pressure when throttle close to 100%
5		Reduce discharge pressure when throttle close to 0%
6		Reduce Discharge Pressures when Cooler Discharge Temperature close to Maximum
7		Reduce Discharge Pressure when Engine needs to Start
8		Inform Dispatcher when Station #5 Pd close to Customer #1 pressure
9		Inform Dispatcher when Supplier #3 Supply pressure close to Dominion pressure
10		Inform Dispatcher when Customer #2 Delivery pressure close to Dominion pressure

Table 2

APO: Automated Pipeline Optimization
MVC Control Software Variables List
24JUN2005

Variables List 2

1-D Array Variables:	Description:
SO_i	0: Station is Bypassed 1: Station has at least one unit Online on a loop controlled by MVC
$ETHP_s$	Total available HP of units that are running now at station
$NTHP_s$	New Total HP computed by Transition when a Compressor needs to Start
$All\_HP\_On_s$	0: There is a future Transition that will require additional net HP to come online 1: No future HP increase is required
ON_i	0: Loop I is off 1: If MVC is running, then MVC is controlling Loop I
UOC_i	Unit Charge Cushion in Hours for lowering DPT_i when a compressor needs to start. $(UOC_i = LIMT_{4i} / 24)$
CST_i	Compressor Start Time (Datetime format) as computed in Transition Logic
DP_i	Discharge Pressure Value at Controlled Station
$DPMT_i$	Minimum Allowable MVC Pressure Target (OPLO from Engineer Screen 1)
DPT_i	MVC Pressure Limit set by Dispatcher on MVCOC (MVC Operating Console)
$DPTC_i$	Adjusted MVC Pressure Target sent to Loop Controller
MOP_i	Maximum Operating Pressure (get from existing ONGDISP_MVCVAL1-6) at Controlled Station
CT_i	Cooler Outlet Temperature
CTL_i	Cooler Outlet Temperature Limit that overrides primary station controller setpoints
SP_i	Suction Pressure Setpoint at manipulated station
$DPSN_i$	Discharge Pressure Setpoint at upstream end of control loop when all down stream stations are in Normal Mode
$DPSE_i$	Discharge Pressure Setpoint at upstream end of control loop when the down stream station needs to start an engine
DPS_i	Most recent Discharge Pressure Setpoint that was sent to compressor station at upstream end of control loop
SD_i	Speed Driver at controlled station (upstream)
$SP_{error-limit_i}$	Maximum error allowed between SP_i and $NEWSP_i$ before EVENT is declared
$NEWSP_s$	Next desired suction setpoint as calculated on Engineer I screen
HP_A_s	Available HP at the station computed by summing the EMM computed HP for each unit that is now available in SCADA
HP_R_s	Required HP for the station as computed by Black Box first iteration after Dispatcher switches "Dispatcher Request MVC" = "ON"
R_s	R_s = 1 if Station is running (Flow Setpoint from SCADA > 0). Applies only to Relay Loop in multi loop Stations

2-D Array Variables:	
$MULT_i$	Multiplier for DPTC adjustment (j > 0) or Expected average rate of change for Station #1 Discharge Pressure while FT is increasing (PSI / Minute) for Flow Target Adjustment (j = 0)
EXP_i	Exponent for DPTC adjustment or Flow Target Adjustment
$LIMT_i$	Limit used in Flow Target Adjustment & Compressor Start Adjustment
$THRSLD_i$	Threshold P, Q, or SD at which X_i is computed
n_i	Damping factor: n is one less than the number of time cycles X_{ij} is averaged over (0 <= n < 100)
$X0_i$	jth Adjustment factor for ith Control Loop at time n=zero (now)
X_{ij}	Table of n historical adjustment factors used for each of i control loops (n dimensioned from 0 - 99)

[0050] Once the various adjustment factors have been determined and stored in the array or other data structure, the system proceeds to determine which of the adjustment factors should be applied. The general process of applying the adjustment factors is shown in the schematic flow diagram of FIGURE 14. In general, an adjusted discharge pressure target is calculated by adding a negative correction equal to the largest (e.g., most negative) of the adjustment factors to a discharge pressure limit which is the highest pressure desired in the pipeline. A time averaging technique is used to smooth the response. If the calculated adjusted pressure target is less than minimum allowable pressure, the calculated adjusted pressure target is set equal to the minimum allowable pressure. The adjusted pressure target

5   is output to the multivariable controller, and the process continues by once again calculating a

set of adjustment factors for the next cycle.

[0051] The foregoing describes the process of calculating ten adjustment factors, and,

for ease of description, describes the process in certain respects as using one- and two-

dimensional arrays or other data structures. However, as those of skill in the art will realize,

10   each of the ten adjustment factors would generally be calculated for a plurality of points

along the pipeline. For example, in the exemplary test system described above with six

compressor stations, the ten adjustment factors may be calculated for each and every one of

the compressor stations. Therefore, the array or data structure used to contain the data may

be multidimensional. As shown by the index subscripts on the variables shown in the figures

15   and described in TABLE 1, certain of the variables are expected to be multi-dimensional,

although in a sufficiently simple system (e.g., one compressor station), they need not

necessarily be multidimensional. Embodiments of the invention may also store past

adjustment factors to use as predictive data for future calculations, for calibration, or for other

purposes.

20   [0052] As was noted briefly above, although there are ten adjustment factors in this

example, embodiments of the invention may calculate and take into account any number of

adjustment factors.

[0053] Software, systems and methods according to embodiments of the invention are

also configured to handle linepack transitions, which are changes in the amount of gas or

25   liquid flowing in the pipeline. These linepack transitions are typically caused by changes in

demand, although they need not be. (Those of skill in the art will note that some of the upset

conditions described above, in effect, deal with linepack transition.) The linepack transitions

may be predicted by the system by accessing past flow data and examining patterns in

19

5   demand over time, or the system may be given an explicit warning by a human operator and

the procedures initiated manually.

[0054] In general, the pseudocode or set of tasks involved in linepack transitions is as

follows:

a.   If pack increase:

10   Compare maximum power of compressors running now to maximum

power of compressors proposed by the energy minimization module output (EMMO).

Using lowest horsepower configuration, compute minimum thru put

and fuel usage when pumped from average of beginning and ending suction pressure

to average of beginning and ending discharge pressure.

15   Determine time and quantity of fuel used to increase pack to correct

level.

Run compressor station at this fuel rate until determined time.

b.   . If pack decrease:

Compare maximum power of compressors running now to maximum

20   power of compressors proposed by EMMO.

Using highest horsepower configuration, compute maximum thru put

and fuel usage when pumped from average of beginning and ending suction pressure

to average of beginning and ending discharge pressure.

Determine time and quantity of fuel used to decrease pack to correct

25   level.

Run compressor station at this fuel rate until determined time.

c.   After Pack is set to correct level at each running compressor station as

computed above, run compressor at flow rate and fuel usage level as determined by steady

state EMMO solution.

· Create EMMO Now and EMMO Future data bases

Use only compressor configuration of EMMO Now and Future

databases in calculating time required.

[0055] More specifically, FIGURE 18 and FIGURE 19 show steps for saving a

revised current nomination and a future nomination, respectively. FIGURE 20, FIGURE 21,

FIGURE 22, FIGURE 23, FIGURE 24, and FIGURE 25 are flow charts illustrating a

transition process according to an embodiment of the invention.

[0056] Embodiments of the invention may also provide a number of administrative

tools, as shown in FIGURE 3. The administrative tools are useful in monitoring the

performance of the system and in calibration of the models. An outline of the functions of

the administrative tools is described below.

[0057]   Tools used to monitor the performance of the system include (i) Fuel

Attainment and (ii) Dispatcher Reporting. Fuel Attainment is an off-line tool that uses

combinations of planned nomination data, system capabilities, historical measured data and

pipeline models to determine the calculated fuel usage for the pipeline for given time periods.

Results of the Fuel Attainment combinations can be compared to actual measured fuels and to

each other to determine system efficiencies and root causes for inefficiencies.

Fuel Attainment combinations include the following named results:

(i) Plan Optimum – Optimized amount of fuel that would have been calculated if full

capabilities were available.(all Stations and Compressors are available for selection)

(ii) Plan – Optimized amount of calculated fuel given planned nomination data and available

system capabilities at the time the optimization occurred.

5 (iii) Operational Optimum – Optimized amount of calculated fuel holding compressor

configurations to their planned nomination settings, station targets to their respective

maximum system capabilities, and using actual data for all other model inputs.

(iv) Flow Variance – Same as Operational Optimum while holding all flows external to the

pipeline model constant at their planned rate.

10 (v) Temperature Variance – Same as Operational Optimum while holding all temperatures

used in the model to their planned values.

(vi) Pressure Variance – Same as Operational Optimum while holding all inlet and outlet

pressures used in the model to the planned values.

(vii) Pressure Target Variance – Same as Operational Optimum while holding all station

15 targets to the planned values.

(viii) Units Used Variance – Same as Operational Optimum while holding all compressor

usage to the planned values.

[0058] Dispatcher Reporting is an on-line tool that uses real-time and historical data,

system capabilities along with pipeline models to determine the current calculated fuel

20 efficiencies for the pipeline. This tool continually publishes current and summary shift

efficiencies for dispatchers for display on SCADA terminals.

[0059] FIGURE 33, FIGURE 34, FIGURE 35, and FIGURE 36 are flow charts

illustrating steps of a method for accurately measuring the fuel savings of the system

according to an embodiment of the invention. Because of the various deliveries, supplies,

25 and operating conditions, historic performance indicators such as fuel per volume pumped

may be inappropriate to accurately measure fuel savings. Rather, a more accurate

measurement may be obtained by dividing the optimal (calculated) fuel + transition fuel by

the actual fuel used.

[0060] Additionally, accurately measuring fuel savings includes taking account of operational variances, such as when inputs into a forecasted optimization run vary from the actual inputs. For example, it is important to capture the fuel costs when a compressor goes down or is not available to run in the optimal solution. Costs may then be quantified to help determine whether to proceed with additional maintenance programs or capital expenditures. FIGURE 37 shows fuel attainment variances that may be taken into account. Table 3 below provides definitions for the variances shown in FIGURE 37.

Table 3

Variance Definitions

1. Planned Units Available Variance % = |[i] Plan Optimum - [ii] Plan | / [ii] Plan

2. Plan Attainment = [iii]Operational Optimum / [ii] Plan

3. Fuel Attainment = [iii] Operational Optimum / [0]Actual Fuel Used

4. Flow Variance % = | [iv] Flow Variance - [iii] Operational Optimum| / [iii] Operational Optimum

5. Temperature Variance % = | [v] Temperature Variance - [iii] Operational Optimum | / [iii] Operational Optimum

6. Pressure Variance % = | [vi] Pressure Variance - [iii] Operational Optimum | / [iii] Operational Optimum

7. Pressure Target Variance % = | [vii] Pressure Target Variance - [iii]Operational Optimum | / [iii] Operational Optimum

8. Units Used Variance % = | [viii] Units Used Variance - [iii] Operational Optimum | / [iii] Operational Optimum

[0061] The tool used for calibrating the models is the Automated Calibration Tool, which is designed to give the engineers appropriate data to analyze and calibrate the models.

5   To ensure reliable calibration, the system finds an appropriate steady state interval for

calibration by ensuring that the pipeline data is sufficiently static and unchanging over a

programmed number of consecutive hours. For example, the initial program may download

actual data after automatically finding 6 consecutive hours of static operating conditions.

Calculations relating to the models' data are then compared to actual data. These values are

10   stored and the engineer is notified when the values fall outside of designed limits.

[0062]   Included in the Automated Calibration Tool is:

(i) Segment Calibration – Compares model results to actual results for line segments. The

parameters used to calibrate the model can include, but is not limited to:

Efficiency

15   Roughness

Heat Coefficient

Ground Temperature

Gas Temperature

(ii) Station Calibration - Compares model results to actual results for stations. Stations are

20   comprised of differing compressor unit types and numbers. The calibration procedure will be

a combination of station and unit level configurations. The parameters used to calibrate the

model can include, but is not limited to:

Inlet Pressure drop

Outlet Pressure drop

25   Temperature drop through coolers

Reciprocating units:

Swept volume

Compression Ratio

Maximum and minimum clearance

Maximum and minimum horsepower

Heat rise coefficient

Fuel

Volume

Mechanical efficiency factor

Compressor efficiency factor

Centrifugal units:

Maximum and minimum horsepower

Head curve

Mechanical efficiency factor

Heat rise coefficient

Fuel

Volume

(iii) Model Variance – Non-optimized amount of calculated fuel given actual system

utilization, measured data, and pipeline models. Pipeline models for Model Variance are

connected, unlike Station and Segment Calibration, which considers station and pipeline

segments in isolation. The formula for Model Variance is:

Mvar(Nom) = Model(Nom,SystemUtil(Nom),Mdata(Nom) – FuelActual(Nom)

Where:

Nom   – Time-period between transient conditions.

SystemUtil(Nom)   – Actual System Utilization during the nomination. (i.e. compressor

statuses, station utilization, etc.

Mdata(Nom)   – Average Measured data during the nomination.

Model(Nom,SystemUtil(Nom),Mdata(Nom)) – A function that calculates unoptimized fuel

usage for a given pipeline, nomination, system utilization, and measured data.

5   Mvar(Nom)   – Calculated Model Variance during a nomination.

[0063] FIGURE 26 shows general calibration processes that may be completed each

day as a scheduled event or run on demand.   FIGURE 27, FIGURE 28, FIGURE 29,

FIGURE 30, FIGURE 31, and FIGURE 32 are schematic flow diagrams illustrating tasks

10   involved in calibrating the system. Model, segment and station calibration processes store

streams of calculated data that may be compared to actual measured data, thereby allowing a

determination to be made whether a specific model is properly calibrated. When a section of

calculated and actual data streams are within a defined tolerance, the model is within

calibration limits for that particular time-period.

15   [0064] Model validation is the process of automatically comparing streams of actual

measured data to streams of calculated data. This process uses predefined validation

parameters for each stream pair to: (i) validate calculated data; (ii) create alarms for

calculated data that is out of tolerance; and (iii) create a calculated-data archive that may be

graphed later by the user.

20   [0065] Validation may be configured to allow any actual measured data stream to be

compared to any calculated data stream using either an absolute or percentage tolerance.

[0066] Validation using absolute tolerance is successful when :

$$0 < (\sum_{n=First}^{n=Last} | \text{Stream}_A [n]_{Calculated} - \text{Stream}_B [n]_{Actual} |) / (\text{Last-First}) < \text{AbsoluteTolerance}_j$$

25   Where Last is the most recent hour of the data stream to be validated and First is the oldest

hour of the data stream to be validated.

30   [0067]   Validation using percentage tolerance is successful when :

5   Percentage Tolerance Successful when:

$$0 < (\sum_{n=First}^{n=Last} |\text{Stream}_A[n]_{Actual} / \text{Stream}_B[n]_{Calculated}|) / (\text{Last-First}) < \%\text{Tolerance}_j$$

10 Where Last is the most recent hour of the data stream to be validated and First is the oldest

hour of the data stream to be validated.

[0068] Tasks involved in the calibration include those shown in FIGURE 27,

FIGURE 28, FIGURE 29, FIGURE 30, FIGURE 31, and FIGURE 32 and described above in

the pseudocode. Data from each segment of the pipeline is examined, particularly those

15 factors identified above in the pseudocode, and the models are corrected using the actual

measured data from the pipeline.

[0069]   The Automated Calibration Tool is not limited to fluid pipeline calibration.

Rather, the calibration process may effectively used wherever models (steady-state or

transient) are employed. Further, the Automated Calibration Tool may significantly reduce

20 the effort required to calibrate a system. For example, the Automated Calibration Tool

utilized nearly 300 model inputs. As a rough estimate, it may take approximately 1 hour per

input to collect data and perform the necessary calculations. Hence, an engineer would take

roughly 300 hours to tune the models once. However, the Automated Calibration Tool

performs these calculations on a daily basis. This significantly reduces man hours required to

25 calibrate models and helps ensure that the models are running safely and efficiently.

[0070] As noted above, pipeline flow optimization systems, software, and methods

may emulate transient optimization by: (i) maintaining the optimal steady-state

configurations and properly adjusting the optimal steady-state pressure targets through upset

conditions employing "Black Box" logic; (ii) identifying through "Black Box" logic, existing

30 field compressor logic, or supply or delivery change of the predictive models, when a new

5   compressor configuration may be required; (iii) transitioning to the new optimal solution as

quickly as possible; (iv) ensuring that the results are accurate through the verification process

of the Automated Calibration Tool; and (v) quantifying the results of the pipeline

optimization process through the Fuel Attainment process. Embodiments of the invention

may provide for significant fuel reductions in compressor stations, thereby resulting in

10   significant fuel savings while operating a pipeline.

[0071] Although the invention has been described with respect to certain

embodiments, the embodiments described herein are intended to be exemplary, rather than

limiting. Modifications and variations will occur to those of ordinary skill in the art, and may

be made without departing from the scope of the invention, which is reflected in the

15   following claims.

WE CLAIM:

    1.   A pipeline flow optimization system, comprising an upset condition handler adapted to provide a multivariable controller with pipeline flow adjustments to handle one or more upset conditions.

    2.   The pipeline flow optimization system of claim 1, further comprising:

    a linepack transition handler adapted to calculate and implement transitions in linepack in a pipeline system; and

    administrative tools,

    wherein the administrative tools are adapted to monitor the performance of the pipeline system and provide data used to calibrate models used for steady-state and transient modeling of the pipeline system.

    3.   The pipeline flow optimization system of claim 2, wherein the transitions in linepack are from a present optimal state to a future optimal state.

    4.   The pipeline flow optimization system of claim 1, wherein the one or more upset conditions comprise one or more conditions selected from the group consisting of an intake flow less than a target intake flow, a discharge pressure greater than a target discharge pressure, the discharge pressure at about a maximum operating pressure, a speed driver of a compressor at about 100%, a speed driver of a compressor at about 0%, a cooler temperature close to maximum, and flow conditions greater or less than contractual obligations at a particular discharge point.

5   5.   The pipeline flow optimization system of claim 1, wherein the upset condition

handler:

            calculates one or more adjustment factors indicative of a degree to which a

correction for a specific upset condition is required;

            determines which of the one or more adjustment factors is to be applied; and

10   supplies an adjustment signal to the multivariable controller based on the one or more

applied adjustment factors.

            6.   The pipeline flow optimization system of claim 2, wherein the administrative

tools provide the data after finding consecutive hours of static operating conditions within the

15 pipeline system.

            7.   The pipeline flow optimization system of claim 6, wherein the administrative

tools provide the data after finding six consecutive hours of static operating conditions.

20   8.   The pipeline flow optimization system of claim 2, wherein the administrative

tools monitor the performance of individual line segments and individual stations and provide

data used to calibrate models used for steady-state and transient modeling of the individual

line segments and individual stations.

25   9.   The pipeline flow optimization system of claim 8, wherein the administrative

tools monitor the performance of coupled line segments and stations and provide data used to

calibrate models used for steady-state and transient modeling of the coupled line segments

and stations.

10.   A pipeline flow optimization system, comprising:

an upset condition handler;

a multivariable controller for controlling components of a pipeline system;

optimization software for calculating steady-state parameters of the pipeline system

and providing the steady-state parameters to the upset condition handler; and

a supervisory control and data acquisition system for acquiring pipeline system data,

wherein the upset condition handler receives the pipeline system data from the

supervisory control and data acquisition system and provides the multivariable controller

with pipeline flow adjustments to handle one or more upset conditions.

11.   The pipeline flow optimization system of claim 10, wherein the upset

condition handler is included within the multivariable controller.

12.   The pipeline flow optimization system of claim 10, wherein the one or more

upset conditions comprise one or more conditions selected from the group consisting of an

intake flow less than a target intake flow, a discharge pressure greater than a target discharge

pressure, the discharge pressure at about a maximum operating pressure, a speed driver of a

compressor at about 100%, a speed driver of a compressor at about 0%, a cooler temperature

close to maximum, and flow conditions greater or less than contractual obligations at a

particular discharge point.

13.   The pipeline flow optimization system of claim 10, wherein the upset

condition handler:

calculates one or more adjustment factors indicative of a degree to which a

correction for a specific upset condition is required;

31

determines which of the one or more adjustment factors is to be applied; and

supplies an adjustment signal to the multivariable controller based on the one or more

applied adjustment factors.

14.   The pipeline flow optimization system of claim 10, wherein the upset

condition handler provides a warning to an operator in response to one or more upset

conditions.

15.   The pipeline flow optimization system of claim 10, further comprising

administrative tools, wherein the administrative tools are adapted to monitor the performance

of the pipeline system and provide data used to calibrate models used for steady-state and

transient modeling of the pipeline system.

16.   The pipeline flow optimization system of claim 10, further comprising a

linepack transition handler adapted to calculate and implement transitions in linepack in the

pipeline system.

17.   A pipeline flow control system, comprising:

an optimizer for calculating steady-state parameters of a pipeline system;

a control unit for controlling pipeline system components that cause pipeline flow;

a data acquirer for acquiring pipeline system component data and pipeline flow data;

and

a control logic unit,

wherein the control logic unit adjusts the steady-state parameters of the pipeline

system in response to an upset condition by analyzing the pipeline system component data

and the pipeline flow data and applying an adjustment factor to the control unit.

18.   The pipeline flow control system of claim 17, wherein the control logic unit

comprises:

an upset condition handler for analyzing the pipeline system component data and the

pipeline flow data; and

a multivariable controller for controlling the pipeline system components that cause

the pipeline flow.

19.   The pipeline flow control system of claim 18, wherein control logic unit

further comprises a linepack transition handler for calculating and implement transitions in

linepack in the pipeline system.

20.   The pipeline flow control system of claim 17, further comprising

administrative tools for monitoring performance of the pipeline system and providing data

used to calibrate models used for steady-state and transient modeling of the pipeline system.

21.   The pipeline flow control system of claim 17, wherein the upset condition

comprises one or more conditions selected from the group consisting of an intake flow less

than a target intake flow, a discharge pressure greater than a target discharge pressure, the

discharge pressure at about a maximum operating pressure, a speed driver of a compressor at

about 100%, a speed driver of a compressor at about 0%, a cooler temperature close to

maximum, and flow conditions greater or less than contractual obligations at a particular

discharge point.

22.   The pipeline flow control system of claim 17, wherein the control unit and the

data acquirer are an integrated unit.

23.   A method of handling transient conditions in a pipeline flow optimization

system, comprising:

calculating one or more adjustment factors, each adjustment factor being

indicative of a degree to which a correction is needed for an upset condition;

choosing at least one adjustment factor to apply to the pipeline flow

optimization system; and

applying a chosen adjustment factor.

24.   The method of claim 23, wherein applying the chosen adjustment factor

comprises automatically adjusting control parameters of the pipeline flow optimization

system.

25.   The method of claim 23, wherein applying the chosen adjustment factor

26.   The method of claim 23, further comprising:

prior to calculating the one or more adjustment factors,

determining whether to adjust a flow target; and

initializing a data array,

wherein storing the one or more adjustment factors comprises storing adjustment

factors in the data array.

27.   The method of claim 23, further comprising storing the one or more

adjustment factors.

FIGURE 1

DELIVERY   △   SUPPLY   ●   Station   ▶   Station Control Logic

FIGURE 2

3/37

FIGURE 3

FIGURE 4

5/37

```
 ┌─────────────────┐
 │      1.1        │
 │  Initialization │
 │  First Iteration│
 └─────────────────┘
          │
          │
 ┌─────────────────────┐
 ├─────────────────────┤
 │First Iteration Switch = OFF│
 ├─────────────────────┤
 └─────────────────────┘
          │
          │
 ┌─────────────────────────┐
 ├─────────────────────────┤
 │Initialize All Variables As Required│
 ├─────────────────────────┤
 └─────────────────────────┘
```

FIGURE 5

Note: Run each time SCADA Execution Counter refreshes. After results are obtained, update APO interface screen.

1.2
Black Box Logic
Adjust Discharge Pressure Targets going into MVC Module

1.2.a
Adjust Flow Target

Initialize X_j Array; X_3 = X_4 ... X_7 = X_8 = 0
Initialize XO_j Array; XO = 0
Initialize DPTC_j Array; DPTC = DPT
I = 0

t = Int(I /2)
s = Int((I + 3) /2)

Odd?
No / Yes

1.2.0
Compute adjustment factors for control loop I

1.2.1
Apply adjustment factors for control loop I

i = i+1

I < 8
Yes / No

J = 7

J < 10
Yes / No

J = J + 1

1.2.0.a
Display Warnings

1.3
Apply MVC Custom Code

FIGURE 6

1.2.a
Adjust Flow Target

Station #2 Is Running

Yes → I = 0

No → I = 1

TN < TS - THRSLD$_{ts}$ / 24

Yes → FPT > Prior FT

No → FT = FPT

FPT > Prior FT

Yes → TE = TS - THRSLD$_{op}$ / 24
TB = TE - (FPT - Prior FT) / MaxR$_{op}$ / 1440

No → FT = FPT

TB < TN < TE

Yes → FT = Prior FT + (FPT - Prior FT) ^ exp$_{op}$ * (TN - Prior TN) / (TE - Prior TN)

Return

Notes:
Unit of time is Days
MaxR is maximum ROC (Rate of Change) of Station #1 Discharge pressure
(psi / Min) Thrsld is Lead Time meaning the time delay from when suction
pressure is adjusted until effect seen at Station #1 Discharge (Hours)

Notes:
Prior FT is the value of FT during the prior iteration
Prior TN is the value of TN during the prior iteration

FIGURE 7

J=4: Reduce Discharge Pressures when throttle close to 100%

When throttle is close to 100%, lower discharge pressure target so down stream station speeds up

Reason: When speed driver is close to 100%, Ps is constrained and cannot go down much. Lowering discharge pressure target will reduce speed driver allowing Ps to control upstream pressure.

J=5: Reduce Discharge Pressures when throttle close to 0%

When throttle is close to 0%, lower discharge pressure target so down stream station speeds up

Reason: When speed driver is close to 0%, Ps is constrained so that Pd cannot go down much. Lowering discharge pressure target will add more room to downstream piping allowing more reaction time if downstream station crashes.

J=6: Reduce Discharge Pressures when Cooler Temperature close to Maximum

When cooler temperature close to maximum, lower discharge pressure target considerably so down stream station speeds up

Reason: Reducing the discharge pressure will reduce the compression ratio which will lower the discharge temperature.

J=7: Reduce Discharge Pressure when Engine needs to Start in a station that is already running

If an engine needs to start and discharge pressure is above the limit for starting an engine, reduce the discharge pressure in time to start the engine

Reason: Station Controller will not allow an Engine to start if Discharge Pressure is higher than the Limit for Starting Engines. This Limit is typically lower than the Pressure held by MVC during normal Operations

1.2.1
Apply adjustment factors

FIGURE 8

1.2.0
Compute adjustment factors for control loop I

J=1: Compensate for Station #1 Flow less than Target

Lower discharge pressure target at Station #1 when total flow into Station #1 < Flow Target

Reason: Create a "hole" in pipeline downstream of Station #1 so (1) there will be room in the pipeline when make-up gas is pumped, and (2) effect on downstream compressor loading will be minimized.

J=2: Compensate for upstream station in Transient condition

When Pd > Pd-target, lower Pd-target at downstream station(s)

Reason: This will cause downstream compressors to speed up faster so that excessively high pressures are minimized.

J=3: Move Station Discharge Pressures away from MOP

When Pd is close to MOP, lower Pd-target considerably

Reason: Then downstream station will pull pressure down quickly.

FIGURE 9

FIGURE 10

FIGURE 11

1.2.0.2
Compute all adjustment factors for all control loops cont'd

j='7: Reduce Discharge Pressure when Engine needs to Start

ETHP$_i$ < NTHP$_i$

Yes → ALL_HP_On$_i$ = 0

No → ALL_HP_On$_i$ = 1

SQ$_i$ = 1
&
ALL_HP_On$_i$ = 0
&
DPSE$_i$ < DPT$_i$

Yes:
dP_Rate = THRSLD$_i$
Ramp_Time = (DPT$_i$ - DPSE$_i$) / dP_Rate / 1440
UCC$_i$ = Limit$_i$ / 24
Begin_Time = CST$_i$ - UCC$_i$ - Ramp_Time

No:

TN > Begin_Time

No → XQ$_i$ = 0

Yes:

TN < Begin_Time + Ramp_Time

Yes:
XQ$_i$ =
- MULT$_i$*(DPT$_i$ - DPSE$_i$) * ((TN - Begin_Time) / Ramp_Time) ^ EXP$_i$

No:
XQ$_i$ =
- MULT$_i$*(DPT$_i$ - DPSE$_i$)

1.2.1
Apply Factors

*Note:
ALL_HP_On = 0 if insufficient HP is running at the
station to meet the future flow target.
ALL_HP_On = 1 if sufficient HP is running at the
station to meet the future flow target.

1.2.0a

Display Warning Message so Dispatcher can make Adjustments Outside of MVC Control

J=8: Inform Dispatcher when Station #5 Pd close to Customer #1 P

Inform Dispatcher when Station #5 Discharge Header #2 is flowing to Customer #1 and Control Valve #2 %open > Limit

Reason: If Customer #1 is receiving gas and Pd at Station #5 goes too low, then Customer #1's requirement cannot be met. Pd on Discharge Header #2 must be raised so contract flow can be met.

J=9: Inform Dispatcher when Supplier #3 Supply Pressure close to Dominion Pressure

Inform Dispatcher when Supplier #3 is Supplying gas to Dominion and Supplier #3 pressure too close to Dominion pressure

Reason: If Dominion is receiving gas and Dominion pressure at Supplier #3 goes too high, then Supplier #3's flow to Dominion cannot be met. Pressure on Pipeline must be lowered so contract flow can be met.

J=10: Inform Dispatcher when Customer #2 Delivery Pressure close to Dominion Pressure

Inform Dispatcher when Dominion is supplying gas to Customer #2 and Customer #2 pressure too close to Dominion pressure

Reason: If Customer #2 is receiving gas and Pipeline pressure at Customer #2 goes too low, then Dominion's flow to Customer #2 cannot be met. Pressure on Pipeline must be raised so contract flow can be met.

1.2.1

Apply adjustment factors

FIGURE 12

FIGURE 13

Note: Purpose of X array is to smooth results over a time average. At this time, seven factors are computed for adjusting discharge pressure targets (j=1-7. Each time step, the most negative factor (the most significant) is selected. A separate n is used for each factor for each control loop for time averaging because the different factors require different amounts of time averaging. n is the number of time steps include in the time average.

1.21
Apply adjustment factors

$SO_i = 1$

No

Yes

$X_{mGJ} = Min(XO_{1,per} \dots XO_{j,per})$

J-picked is the j selected above that minimized X_{mGJ}

last = $n_{i,J\text{-}picked}$

$DPTC_i = DPT_i + (X_{mGJ} + \dots + X_{J\text{-}last})/(n_{i,J\text{-}picked} + 1)$

$DPTC_i < DPMT_i$

Yes

$DPTC_i = DPMT_i$

1.20
Compute adjustment factors for next control loop

FIGURE 14

```
┌─────────────────┐
│      1.2.2      │
│ Black Box EVENT │
│      Logic      │
└─────────────────┘
```

When MVC is turned on:
  Sum available HP of all units that are running at each station (HP_Aa) (See 1.1: MVC Initialization).

Each time MVC runs:
  Check each running compressor station as follows:
  (A): If [HP_Aa < HP_R₁ (As calculated by Black Box Initialization) AND (SP₁ – NEWSP₁) > SP error-limit, THEN turn Event Mode on at that station only.
  (B): When MVC enters Event Mode at a station:
        1. Send message to operator stating which station is in event.
        2. Set MVC button at bottom of SCADA Screen to a unique color, such as orange.
        3. Set background color of Station Name in Event on TL-400 APO Interface Screen to a unique color, such as orange.
        4. Set Pd-setpoint at station in Event to DPSE₁.
        5. Set Pd-setpoint at all upstream stations to the greater of DPSE₁ or (DPSN₁ – Event_Offset₁.)
        6. Set INT SPHi limits on MVC Engineer 1 screen to Pd-setpoints set above.

When in Event Mode:
  (A): Disregard low limit on Engineer 1 screen & disable 100% Speed Driver Logic at station in Event so a unit will start.

When station reverts back to Normal mode:
  (A): Reset Pd-setpoints and INT SPHi limits to DPSN₁.
  (B): Send message to operator stating which station is back in Normal mode.
  (C): Reset colors to Normal.

FIGURE 15

FIGURE 16

**1.3.1
MVC EVENT Mode
Logic**

When in Event Mode:

(A): Disregard low limit on Engineer 1 screen & disable 100% Speed Driver Logic at station in Event so a unit will start.
(B): When EVENT clears, start a timer (initially set at two Black-Box cycles) tuned to allow Ps to stabilize
(C): After timer times out, Revert 100% Speed Driver Logic to Normal

MVC Module
Display Factors on an Engineer's screen
Apply Appropriate Discharge Pressure Setpoints

FIGURE 17

1.0
Save Revised Current Nomination

Dispatcher runs EMMI for revised current flow or change in available facilities.

Dispatcher reviews and approves EMMO results.
Dispatcher makes note of Facilities changes hitted on EMMO and immediately begins to implement changes manually.

Dispatcher saves Current Flow Target:
Dispatcher clicks on "Save Nomination" button
Dispatcher Clicks "Current Nomination Type"
Dispatcher clicks "Save"

APO resets values on APO Interface screen:
Flow Target New = new value
Future Flow Target removed
Time Start removed

FIGURE 18

2.0
Save Future Nomination

Dispatcher runs EMMI for future nomination amount.

Dispatcher reviews and approves EMMO results.

Dispatcher saves Future Nomination:
Dispatcher clicks on "Save Nomination" button
Dispatcher Clicks "Future Nomination Type"
Dispatcher enters new Effective Future Nomination Date.
Dispatcher clicks "Save"

APO resets values on APO interface screen:
Flow Target Now = new value
Future Flow Target removed
Time Start removed

FIGURE 19

20/37

FIGURE 20

21/37

FIGURE 21

22/37

FIGURE 22

```
          ┌─────────────┐
          │  Determine  │
          │ Transition  │
          │    Times    │
          └─────────────┘
                 │
                 ▼
    ┌──────────────────────────────┐
    │ Balance the LoadCalibration   │
    │ Model to compute the Pump     │
    │ Rates and Transition Fuel     │
    │ Rates                         │
    └──────────────────────────────┘
                 │
                 ▼
         ┌─────────────┐
         │  Station=0  │
         └─────────────┘
                 │
                 ▼
    ┌──────────────────────────────┐
    │   PumpRate(Station)=          │
    │   Balanced(PumpRate(station)) │
    └──────────────────────────────┘
                 │
                 ▼
    ┌──────────────────────────────┐
    │  TransitionFuelRate(Station)= │
    │  Balanced(FuelRate(Station))  │      ┌──────────────────────┐
    └──────────────────────────────┘      │ Station = Station + 1 │
                 │                         └──────────────────────┘
                 ▼
    ┌──────────────────────────────────────┐
    │ NetPackRate(Station)=Flow(Station(Nom1))│
    │ +Fuelrate(Station(Nom1))-Pumprate(Station)│
    │   -TransitionFuelRate(Station)        │
    └──────────────────────────────────────┘
                 │
                 ▼
┌──────────────────────────────────────────────────────────────┐
│ TimeToPack(Station)=|[Lp(Station(Nom1)-Lp(Station)(Nom2)]|/NetPackRate(Station)│
└──────────────────────────────────────────────────────────────┘
                 │
                 ▼
          ┌─────────────┐
          │     TT2      │
          └─────────────┘
                 │
                 ▼
   ┌──────┐         ┌──────────────┐
   │Return│◄── No ──│      If       │─── Yes ──►
   └──────┘         │   Station<    │
                    │  Last_Station │
                    └──────────────┘
```

FIGURE 23

FIGURE 24

FIGURE 25

FIGURE 26

FIGURE 27

FIGURE 28

FIGURE 29

FIGURE 30

Inputparameters:
1. Model that contains all of the CompressorStations of the pipeline. The model is capable of determining steady state conditions for each station when populated with actual data from the pipeline. <Model>
2. A starting date: <StartDate>
3. A ending Date: <EndDate>
InputData:
1. An archive of data containing hourly measured values for the station. I.e. temperatures, pressures, and flows. <Archive>
OutputData:
Hourly Results from the Model when steady state conditions occurred.
1.

FIGURE 31

32/37

Archive Is the name of the Database containing hourly historical SCADA data for measurement points. StationCalibration Is the name of the Database containing calculated hourly data from Station Calibration.	StationItem Is an Enumerated list of attributes common to a station. Such as Pa, Pd, Ta, Td etc. CompressorItem Is an Enumerated list of attributes common to a compressor. Such as Pa,Pd,Ta,Td etc.

Verify Segment Calibration

Start Validate Station Calibration

Create Reference to Archive Databases as <Archive>

Archive Database
Referenced by:
<Archive><DateTime><Station>.Item(StationItem)
<Archive><DateTime><Station><Compressor>.Item(CompressorItem)

Create Reference to Calibration Databases as <Calibration>

Calibration Database
Referenced by:
<StationCalibration><DateTime><Station><Station>.Item(StationItem)
<StationCalibration><DateTime><Station> <Compressor>.Item(CompressorItem)

Create Reference to Station CalibrationAlarm Configuration Databases as <Calibration>

Station Calibration Alarm Configuration Database
A Data structure used to determine when a data item is out of calibration. The structure contains one object or record for each userconfigurable alarm.
Each configurable alarm compares an Archived Scada data item to a Calculated Data Item that was generated and stored from the StationCalibrationAlgorithm.
Referenced as:
<StationCalibrationAlarm>
Each object or record contains configuration for a single alarm with the following elements:

Used to define the Scada Actual Measurement Point
ActualDataItem.Station
ActualDataItem.Compressor
ActualDataItem.StationItem
ActualDataItem.CompressorItem
Used to define the Calculated Measurement Point
CalculatedDataItem.Station
CalculatedDataItem.Compressor
CalculatedDataItem.StationItem
CalculatedDataItem.CompressorItem

Configuration Items
D - Days of data to use
AllowedToleroncePercentage   uses |ItemA - ItemC| >
AllowedTolerance
AllowedTolerance   - uses |ItemA - ItemC| > AllowedTolerance
NoDataAlarm   - maximum number of days without a steady state conditions to allow withoutgenerating an alarm.

For each item in <StationCalibrationAlarm>

Obtain References to one Archive and one Calculated set of data from the Archive and Calibration Databases

Foreach <DateTime>hour in the last <StationCalibrationAlarm>.D Days

(Inner Loop Routine Pseudocode)
difValue = (Archive data value for current the hour) -
(Calculated data value for the Current hour)
difValue =abs(difValue)
If (difValue > AllowedTolerance)
add alarm message to the "Station Calibration Alarm Database"
difValue% = (Archive data value for Current the hour) /
(Calculated data value for the Current hour)
difValue% = abs(100 - difValue)
If(difValue% > AllowedTolerancePercentage)
add alarm message to the "Station Calibration Alarm Database"

store "difValue" and "difValue%" to the "Station Calibration Alarm Database" for trending later by a graphical tool.

Loop until the last Hour

Station Calibration Alarm Database
Stores generated Alarms that can be stored using the following functions as:
<StationCalibration>(<Station>,<DateTime>,<StationItem>,Message)
<StationCalibration>(<Station>,<Compressor>,<DateTime>,<CompressorItem>,Message)
Stores the Trends that are used to determine if an alarm should be generated using:
<StationCalibration>(<Station>,<DateTime>,<CompressorItem>,doubleValue)
<StationCalibration>(<Station>,<Compressor>,<DateTime>,<CompressorItem>,doubleValue)

Loop until last item in <StationCalibrationAlarm>

Return

FIGURE 32

33/37

Using Actual Conditions in Fuel Attainment Model, compare Theoretical fuel to Actual fuel used over the continuous block of hours when running HP remains constant at all compressor stations. Consider Unepack change from beginning to end of block. Nom1 is beginning of block. Nom2 is end of block.

Note: All Historical pressures, flows, temperatures, etc. are averaged from LargeBlockStart to LargeBlockEnd. Unit configuration is Actual = Nom 1 = Nom2 configuration.

Return A

Determine the Starting and Ending times of the Largest Block of Hours when running HP remains constant at all compressor stations. Then compute average flow, pressure & temperature values for all data required by Base Model. Then load actual compressure configurations & all averaged data into Base Model as described below:

Set the Primary Inlet Suction pressures = suppliers historical delivery Pressure at Station #1
Set temperature at Inlet node = historical temperature.
For Supplies & Deliveries: Use Average Historical Flow

Set system outlet node to NodeType = N

Use the Historical Ambient air temperature to calculate the available horsepower at the centrifugal units by setting the AmbientTemperature in the GasCompressor Table = ambient air temperature at Columbus + k. Set LimitingOutletTemp in CC & RC tables = average cooler temperature from SCADA (24 Hour running average) skipping times when station was bypassed. Set = unique constant = 100 if no cooler. All constants should eventually end up in Engineers calibration table.
Set: Station = 0, TotGnd =0, SumGnd =0.

Compressor Station is running in Nom1 Optimized Model ◄──No── Compressor Station is running in SCADA ──Yes──►

If Station has multiple suction headers

1: In GasNode Table, set NodeType (Suction Node) = QF
2: In GasCompressor Table, set Status = ""
3: In GasCompressor Table, set ValueKUtilized = 5000000

1: Set all compressor STATUSes same as SCADA. Online if running. Offline if not.
2: For each Online Compressor: Add average of Nom1& Nom2 suction temps to TotGnd. Add 1 to SumGnd.
3: Set discharge pressure Node to BF
4: Set Node Pressure = Average Historical Pressure
5: Set compressor station on Discharge Control
6: Set PDMAX in Compressor Station = Average Historical Pressure
7: Set ValveUtilization = 0

If suction node is Type BF ──Yes──►

Set compressor to UNKNOWN

Set suction node to Type = QF

Multiple Suction Headers

Station + 1 ◄─Yes─ If Station < Last_Station ─No──► Ground Temperature = TotGnd / SumGnd ──► Balance Solver (Flag If Infeasible) ──► Page 2
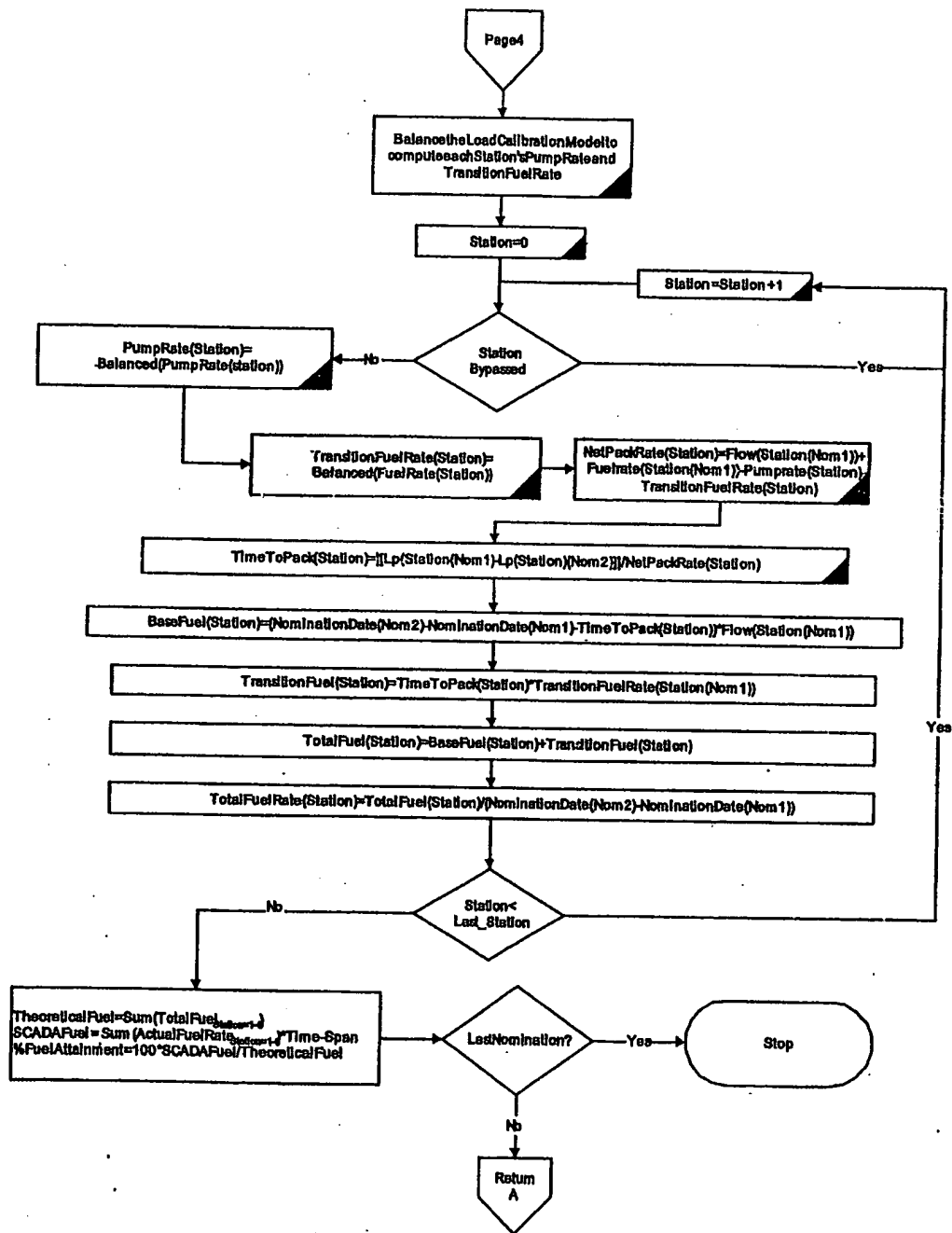
FIGURE 33

FIGURE 34

FIGURE 35

FIGURE 36

FIGURE 37